Demystifying HTML5

Sulamita Garcia, Intel

Gustavo Barbieri, Profusion

Twitter: @Develop4AppUpEu

Demystifying HTML5

- Basics
  - W3C and WhatWG
  - What's new
- Distributing HTML5 Applications
  - Intel AppUp Encapsulator
  - WebKit
- Hybrid Applications
- Talk is cheap, show me the code

Basics

W3C and HTML

OK, but they're not gonna like it!

- "Implementations and specifications have to do a delicate dance together."
- Board overseeing draft
- Dynamic circular work
  - Browsers implement features in draft
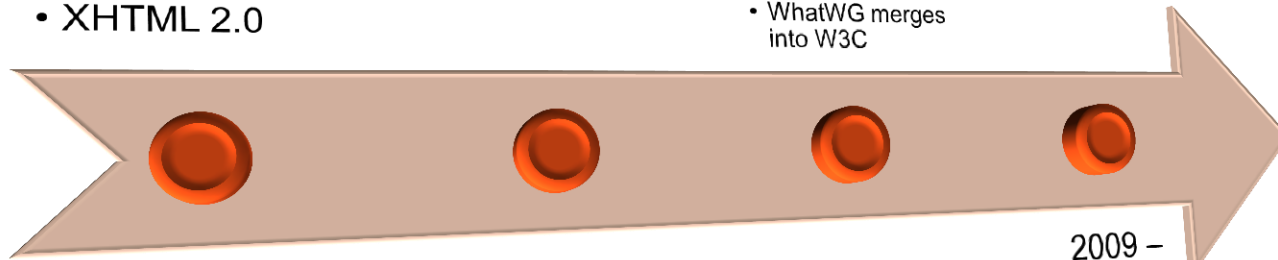 - Draft implements features used in browsers

Enters WhatWG

2004 –
Workshop
- 11 x 8 – not extending HTML & CSS
- XHTML 2.0

2006 – **HTML5**
- WhatWG merges into W3C

2004 –
dissidents
create
WhatWG
- Web Applications 1.0

2009 –
XHTML 2.0
WG shuts
down

What's new - Big picture

- HTML5
  - New tags

- Javascript
  - New libraries

- CSS3

```
<html>
   <head>
         <title>Sample Web Page</title>
         <script>
         var context =
         canvas_area.getContext("2d");
         </script>
   </head>
   <body>
         <canvas id="canvas_area"
         width=300 height=200> </canvas>
   </body>
</html>
```

What's new – new resources

- **<video>, <audio>, <canvas>, <section>, <article>, <nav>, <ruby>, <time>, <header>, <footer>, <progress>…**

- **<input> new types: tel, search, url, email, date, number, color...**

- **Geolocation, offline storage, webgl…**

What's New - Simplifying XHTML

Original

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

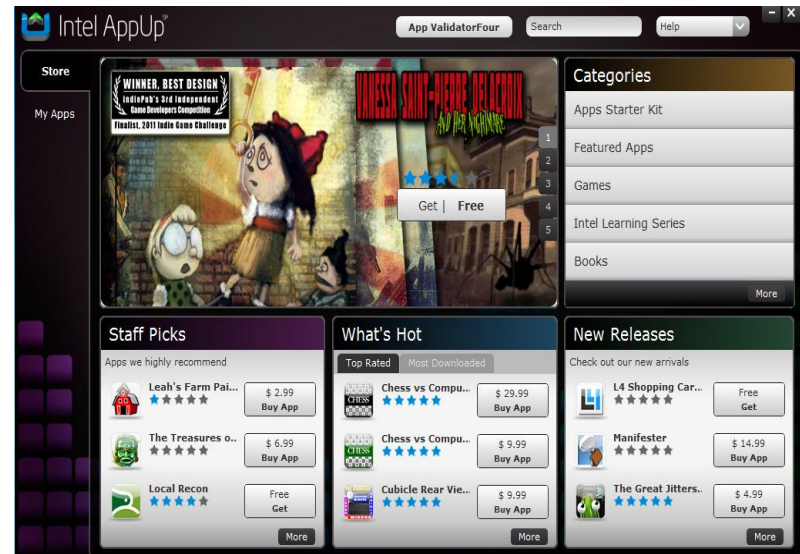<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

HTML5

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8" />

<meta name="robots" content="noindex" />

Basics

- HTML5 Definition is overseen by W3C together with developers and browsers, dinamically

- HTML5 ~= HTML + CSS + JS

- Simplyfying and new tags

Distributing HTML5 Applications

Intel AppUp

- Distribution channel for web applications
  - Plus C/C++, Java, Flash, .Net and Adobe Air

- +30 co-branded stores worldwide

- Open source applications

- Developer Program
  - SDK, documents, plug-ins

- English, French, German, Italian, Spanish

Intel AppUp Encapsulator

- Supports many HTML5 features
- Integrated with AppUp
- Asks for GUI and info
- Provides .msi and .rpm packages

Encapsulator features support

Encapsulator

- Forms

- Storage

- Canvas

- Web Applications...

WebKit

- Engine: what renders code

- Open Source

- Example of other engines: Gecko(Firefox), Trident(IE), Presto(Opera)

- Currently version 2.2

WebKit-EFL

- New port (2009)
- Open since 2010, Upstream since 2011
- Shares with Gtk (Cairo, Soup, Gstreamer...)
- Mainly by ProFUSION and Samsung
- EFL: Evas, Ecore and Edje
- Widget-set independent!
- X11 independent. Runs on FB, DirectFB...
- Unique mobile features:
    - Fast Zooming (Weak Zoom)
    - Fast Scrolling (Tiled Backing Store)
    - Vectorial Zooming (Cairo Scaling)

WebKit-EFL Features

- html5test.com
  - Chrome 14 = 340
  - Firefox 6 = 313
  - WebKit-EFL = 310
- Graphics (Canvas), Video, SVG, CSS...
- Plans for WebKit2 (multi process/threads)
- No NS Plugins (Flash, Java)

Distributing HTML5 Applications - Summary

- Intel AppUp offers a unique distribution channel

- AppUp Encapsulator has support to many features and will continue to improve

- WebKit is an open source engine where many projects use, like Encapsulator and Tizen

Hybrid Applications

Hybrid Application Use Cases

- Easier access to Web2.0 services
- Simplify complex GUI elements
- Portability
- Still integrated (native navigation, etc)
- Tighter Control over Web Runtime
- Examples: Twitter, Facebook, RSS Readers

Hybrid Application Models

- Custom Protocols
- Custom JS Objects
- Native code generating HTML, JS or CSS
- Partial views (widgets) using webviews

Hybrid - Custom Protocols

- Registered by Apps in their web views
- Example: `app://something?parm=1&b=2`
- Used as callback: click, mouse over, JS, …
- Used to generate custom CSS, HTML...

Hybrid - Custom JS Objects

- Registered by Apps in their web views
- Bindings to Native Code
- Properties and Functions
- Good for more complex APIs
- Example:

```
var myObj = window.myObj;
myObj.someFunction(1, "hello",
                   {"key": 2});
```

Hybrid – Code Generated Resources

- Native code creates HTML, CSS, Images...
- Injects to some frame using setContent()
- Example:

```
char *html = generate_html(ctxt);
ewk_frame_contents_set(frame, html...);
```

- ...or returns as custom protocol contents
- Example:

```
myapp://graph2d?x=0&y=0&data=1,1&data=2,2&data=3,3
```

Talk is cheap, show me the code

Detection techniques

- Check if a certain property exists on a global object (such as window or navigator).

```
return !!navigator.geolocation;
```

- Create an element, then check if a certain property exists on that element.

```
return
!!document.createElement('canvas').getContext
;
```

Detection techniques

- Create an element, check if a certain method exists on that element, then call the method and check the value it returns.

```
var v = document.createElement("video");
return v.canPlayType('video/mp4;
    codecs="avc1.42E01E, mp4a.40.2"');
```

- Create an element, set a property to a certain value, then check if the property has retained its value.

```
var i = document.createElement("input");
i.setAttribute("type", "color");
return i.type !== "text";
```

New HTML5 interesting tags

- Canvas

- Geolocation

- Video, Audio

- Section tags

Canvas

```
<canvas id="canvas" width=300 height=300></canvas>

<script>
    var canvas_d = document.getElementById("canvas");
    var ctx = canvas_d.getContext("2d");
<!--- Drawing --->
    ctx.beginPath();
    ctx.moveTo(x,y);
    ctx.lineTo(x,y);
    ctx.strokeStyle = "#eee"
    ctx.stroke();
```
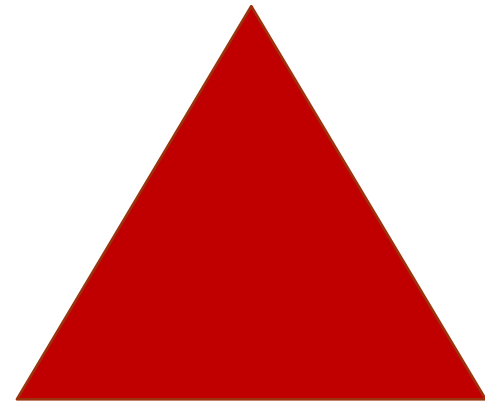
Canvas

```
ctx.beginPath();
ctx.moveTo(x,y);
ctx.lineTo(x+50,y);
ctx.lineTo(x+25,y+50);
ctx.closePath();
ctx.fillStyle = "#ffc821";
ctx.fill();


ctx.beginPath();
ctx.rect(x,y,w,h);
ctx.closePath();
ctx.fill();
```

Canvas animations

- Draw
- Clean
- Move
- Repeat...

```
ctx.clearRect(x,y,w,h);

ctx.restore();
```

# Canvas Animation Example - AltMegaRace

Geolocation

```
navigator.gelocation.getCurrentPosition(show_map, error_f);  <!--- callback --->

function show_map(position) {
        var latitude = position.coords.latitude;
        var longitude = position.coords.longitude;
        var latlng = new google.maps.LatLng(latitude, longitude);
        var myOptions = {
                zoom: 15,
                center: latlng,
                mapTypeId: google.maps.MapTypeId.ROADMAP
        );
        var map = new google.maps.Map(document.getElementById("mapcanvas"),
                myOptions);


        var marker = new google.maps.Marker ( {
                position : latlng,
                map: map,
                title: "You are here!"
        } ) ;
}
```

Geolocation - results
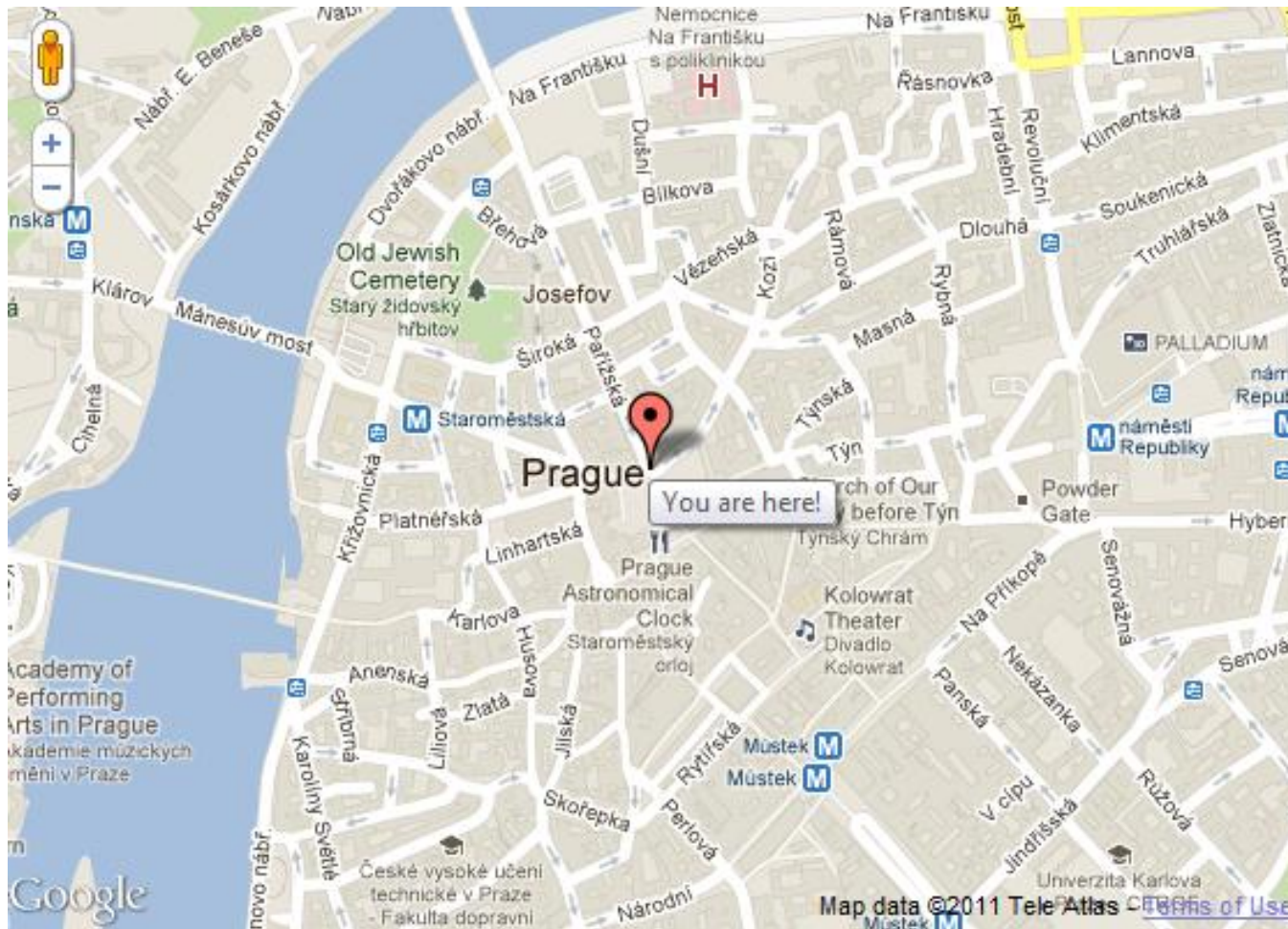
Geolocation - Permissions

Ask for user permission:


```
function error_f() {
     if (err.code == 1) {
             // user said no, show map default location
     } elseif (err.code == 2) {
             // position unavailable
     } elseif (err.code == 3) {
             // timeout
     } else {
             // error unknown (0)
     }
}
```

Video

- It's complicated...
- Codecs and patents making everything difficult
- Supporting <video> not necessarily means supporting MPEG-4 or H.264

```
<video src="video.mp4" width=320 height=240 autoplay> </video>


<video width=320 height=240 controls>
   <source src="video.mp4" type='video/mp4;
   codecs="avc1.42E01E, mp4a.40.2"'>
   <source src="video.webm type='video/webm; codecs="vp8,
   vorbis"'>
   <source src="video.ogv" type='video/ogg; codecs="theora,
   vorbis"'>
</video>
```

Audio

```
<audio src="audio.ogg" controls="controls">
Your browser does not support the audio element.
</audio>


<audio controls="controls">
  <source src="audio.ogg" type="audio/ogg" />
  <source src="audio.mp3" type="audio/mpeg" />
Your browser does not support the audio element.
</audio>
```

Section tags

- Great for syndication and dynamically compounding
- Avoiding confusions with <h1>-<h6>

```
section
|
+--h1 (first heading, child of section)
| |
| +--text node "Hello WebWorld"
|
+--p (child of section, sibling of h1)
   |
   +--text node "This is your text"
```

Section tags

- No more <div>s
- <section>
- <article>
- <aside>
- <header>
- <hgroup>
- <figure><figcaption>
- <nav>
- <footer>

```html
<body>
<p> Some text paragraph, pretend it's long...</p>
<section>
        <h1>This is the first section</h1>
        <p> This is the section text</p>
</section>
```

Section tags – <article>

```
<article>
        <header>
        <h1>How we got here</h1>
        <h2>Hot Topic</h2>
        <h2>Who defines HTML5</h2>
        <h2>A bit of history</h2>
        <nav>
                <ul>
                        <li><a href="#">home</a></li>
                        <li><a href="#">home</a></li>
                        <li><a href="#">home</a></li>
                        <li><a href="#">home</a></li>
                </ul>
        </nav>
        </header>
        <p>Lorem ipsum … </p>
        <footer>
                <p> That's all folks!</p>
        </footer>
</article>
```
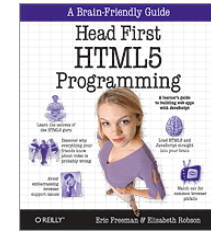
Links

- http://appup.com/
- http://appdeveloper.intel.com/
- http://appdeveloper.intel.com/en-us/article/html5-feature-compatibility-intel-appup-encapsulator-beta
- http://appdeveloper.intel.com/en-us/article/html5-game-development-appup-part-1
- diveintohtml5.info / HTML5 Up and Running
- html5test.com – score/support check
- Head First HTML5
- http://evolutionofweb.appspot.com/
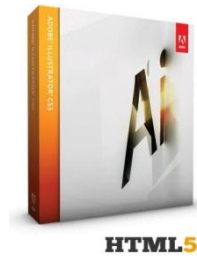- http://html5demos.com/

- Specification: http://www.whatwg.org/html
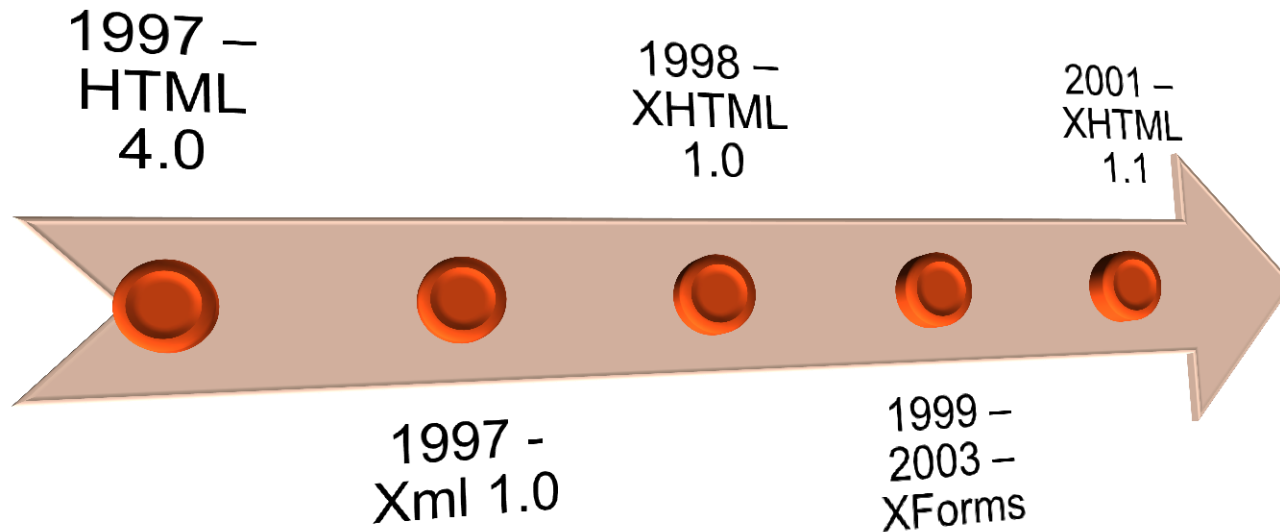- http://en.wikipedia.org/wiki/Comparison_of_layout_engines_(HTML5)

Backup

HTML5 – Hot Topic

- Revolution ongoing
- History seems to have taught stuff
- Intel embracing HTML5

How we got HTML5 – W3C WGs

1997 – HTML 4.0

1998 – XHTML 1.0

2001 – XHTML 1.1

1997 - Xml 1.0

1999 – 2003 – XForms

Forgiving browsers and loopholes kept messing up everything...

Simple HTML page

CSS – Cascading Style Sheets

```
#CSS for our sample webpage

body {
        background-color:#d0e4fe;
}
h1 {
        color:orange;
        text-align:center;
}
p {
        font-family:"Times New Roman";
        font-size:20px;
}
```